

Decision Matrices, Rank, and Information Flow in Neural Networks

1. From activations to decisions

A layer in a neural network is usually written as:

$$z = Wx$$

Applying a ReLU gives:

$$\text{ReLU}(z) = Dz$$

Here, D is the **decision matrix**:

- A diagonal matrix of 0s and 1s
- Chosen based on the input x

So each layer can be rewritten as:

$$x_{\text{out}} = DWx$$

D is doing row selection on W .

Including the weight matrix of the next layer

$$W_2 D_1 W_1 x$$

You can have D act on W_2 instead which is column selection on W_2 .

Then $S = WD$ is a conditional synthetic matrix and a ReLU neural network can be construed as a chain of such synthetic matrices.

This means:

A neural network is not one fixed transformation, but a **sequence of input-dependent matrix choices**.

2. The decision tree viewpoint

Each possible pattern of 0s and 1s in D defines a different linear mapping.

So across layers:

- The network is effectively selecting a path through a huge **decision tree**
- Each path corresponds to a different product of matrices

$$W_{\text{eff}} = S_n \dots S_2 S_1 W$$

This is why neural networks are expressive:

- They can represent many different linear maps
 - The input determines which one is used
-

3. Where rank loss comes from

The problem is that each decision matrix D:

- **Zeros out columns**
- Reduces the number of active dimensions

So each layer can:

- Lower the **rank** of the transformation
- Reduce the amount of information that survives

Once a direction is removed:

- It cannot be recovered later
- Future layers never see it

So along a path in the decision tree:

Information is progressively **discarded**, not just transformed.

4. Two-output idea helps

With two complementary matrices:

$$W_{\text{eff}} = W_1 D + W_2 (I - D)$$

Now:

- No coordinate is zeroed out
- Every component is processed by either W_1 or W_2

This changes the situation:

- The transformation is **dense**
- Immediate rank collapse is less likely
- Information is **routed instead of deleted**

In decision-tree terms:

- Instead of pruning branches, you **redirect them**
-

5. But rank problems still remain

Even without zeroing:

- Columns of W_1 and W_2 can still be **dependent**
- Different paths can converge to similar representations
- Repeated composition can still reduce effective dimensionality

So:

Avoiding zeroing $\neq$ guaranteeing full rank

The system is safer, but not immune.

6. Why information flow matters

Under the decision matrix viewpoint, each layer:

- Makes a **local decision** about how to process data

But later layers:

- Still need access to **earlier information**
- To make better conditional decisions

If information is lost early:

- Later decisions are made with incomplete context
- The decision tree becomes less useful

So good neural networks must ensure:

Information is not just processed—it must **continue flowing forward**.

7. The role of residual connections

Architectures like ResNet use:

$$x_{l+1} = x_l + F(x_l)$$

This guarantees:

- A **direct path** that preserves the input
- A baseline transformation with full rank (identity)
- Protection against cumulative information loss

In decision-tree terms:

- Every branch retains access to the **original signal**
-

8. Putting it all together

- **Decision matrix (D)**: selects which transformation to apply
 - **Standard ReLU**: deletes information (rank can drop)
 - **Two-output ReLU**: routes all information (less immediate loss)
 - **Deep composition**: still risks gradual collapse
 - **Residual connections**: ensure information always survives
-

Final intuition

Think of the network as a sequence of decisions:

- Each layer asks: *“How should I process this?”*
- The decision matrix implements that choice

But for those decisions to be meaningful:

- The network must **retain enough information** to decide well
- And that requires preserving rank and signal flow across depth

So the core principle is:

Neural networks work best when they **combine conditional computation (decision matrices)** with **strong information preservation (residual flow)**.